

4ALL BANK

Any Time Any Place
Any Device Always ON.....

Core Banking Project

Team Number: 13

Submitted To:
Mr. Vijay Gupta
Lecturer
IIM, Jaipur

Made By:
Sheetal Saha
Narendra Kr. Prajapat
Sagar Chaturvedi

CERTIFICATE

This is to certify that this report embodies the original work done by **Sagar Chaturvedi, Sheetal Saha and Narendra Prajapat** during the project submission as a partial fulfillment of the requirement for the **System Design Project** of Masters of computer Application IV semester, of the Rajasthan Technical University, Kota.

Swati V. Chande

Principal

(MCA Department)

International School Of Informatics and Management

Mr. Vijay Gupta

Lecturer

(MCA Department)

International School Of Informatics and Management

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crown all efforts with success.

We are grateful to our project guide Mr. Vijay Gupta for the guidance, inspiration and constructive suggestions that helped us in the preparation and execution of this project.

We would also express our thanks to our Friend Yogesh Jain and people who have helped in successful completion of the project.

**Sagar Chaturvedi
Sheetal Saha
Narendra Prajapat**

Table of Contents

S.NO	TITLE	PAGE NO
1.	SYNOPSIS	
2.	FEASIBILITY REPORT	
3.	PROJECT DESCRIPTION	
4.	DATA FLOW DIAGRAM	
5.	TABLE STRUCTURE	
6.	SOURCE CODE	
7.	FORM LAYOUT	
8.	REPORTS	
9.	CONCLUSION	
10.	REFRENCES & BIBILIOGRAPHY	

SYNOPSIS

Purpose:-

The basic banking suite provides a global accounting foundation that provides the all private banks with electronic banking facilities. It allows the client of private banks to carry out their day to day banking transactions.

Scope:-

The core banking project is widely applicable with private banks. It can even be used in industries for their personal transactions (working).

Technologies:-

- **Hardware Requirements:-**

- ❖ PC with 2GB hard disk
- ❖ 256 MB RAM

- **Software Requirements:-**

- ❖ Windows XP with MS-OFFICE
- ❖ Database- MY SQL/MS-ACCESS
- ❖ Programming Language – VB.NET
- ❖ IDE- Visual Studio 2005

Overview:-

Core banking is all about knowing our customer need and provide them with the right service at the right time through right channel 24*7 day a week. Being “electronic”, it not only provides its customers with faster and better facilities, it even reduces the manual overhead of accounts maintenance.

Functional components of the project:-

Following are the functional needs of the software:-

1. Customer must have a valid user ID and password to login to the system.
2. After the valid user logs in, the system shows the present balance in that particular account number.
3. Customer can perform transactions like deposit and withdrawal from his account.
4. Proper help to be provided as and when requested by the customer.

More functionality can be added to “enhance the project”:-

1. By adding new modules of different accounts like saving A/C, current A/C etc. to facilitate new customers/users.
2. By the use of electronic media, “Digital Signature” on the card can be provided with the customer to make it secure and efficient.

FEASIBILITY REPORT

Understanding Feasibility

Feasibility study means the analysis of problem to determine if it can be solved effectively. In other words it is the study of the possibilities of the proposed system it studies the work ability, impact on the organization ability to meet user's need and efficient use of resources.

Three aspects in which the system has to be feasible are:-

1. ECONOMICAL FEASIBILITY:-

The economical analysis checks for the high investment incurred on the system. It evaluates development & implementing charges for the proposed "Banking Project". The S/W used for the development is easily available at minimal cost & the database applied is freely available hence it results in low cost implementation.

2. TECHNICAL FEASIBILITY:-

This aspect concentrates on the concept of using Computer Meaning, "Mechanization" of human works. Thus the automated solution leads to the need for a technical feasibility study. The focus on the platform used database management & users for that S/W.

The proposed system doesn't require an in depth technical knowledge as the system development is simple and easy to understand. The S/W (VB.NET) used makes the system user friendly (GUI). The result obtain should be true in the real time conditions.

3. BEHAVIOURAL FEASIBILITY:-

Behavioral feasibility deals with the runtime performance of the S/W the proposed system must score higher than the present in the behavioral study. The S/W should have end user in mind when the system is designed while designing s/w the programmer should be aware of the conditions user's. Knowledge input, output, calculations etc. The s/w contains only a minimum no. of bugs. Care should be also taken to avoid non-working means & buttons.

“PRESENT SYSTEM”

The developed system is an innovation in the area of private banking. In the existing system the no. of staff required for completing the work is more, while the new system requires lesser staffs generally. The data entry process requires the data on the paper, which is then feed into the application by the operator while doing so; the data entry operator has to look into the paper again & again and thus the chances of inaccuracies in the typed contents increases. Also the process includes higher transportation cost, increased handling cost, more time delays, low accuracy, more usage of resources like registers, books, papers, etc.

“PROPOSED SYSTEM”

“Why an Automated Private Banking System?”

- ◆ Almost 60% of today’s information is still paper based.
- ◆ 30% of all office time is spent finding documents.
- ◆ The average time to manage a single document is 12 minutes, 9 minutes to re-file and 3 minutes to process.

Hence the requirement is to develop a system that minimizes all these overheads included while giving the maximum output for the organization. The basis for the project is to develop a fully automated banking system that includes depositing of amount, withdrawal of amount and exporting the outcome back to the client while considering all the tools and facilities than a client may need for efficient and effective output.

Benefits of the system

- Quick, authenticated access to accounts via the desktop.
- Easily scalable to grow with changing system requirement.
- Enterprise wide access to information.
- Improved information security, restricting unauthorized access.
- **Minimize Storage Space**

In manual system, much storage space for data files is required so to overcome this problem, an automated well managed database is developed for saving storage space. This s/w saves space and stores information efficiently. It ends the burden of having large manual filing storage system

Banking System can be used extensively

- Withdrawal of amount by the client.
- Deposition of amount by the client.
- Faster balance enquiry.

TABLE STRUCTURE

Account_No	User_Name	Password	Acoount_Bal
1	Sagar	007	1000
2	James	008	2000
3	Sheetal	009	3400
4	Narendra	Prajapat	4000

SOURCE CODE

Module1

```
Module Module1
    Public con As New OleDb.OleDbConnection
    Public ds As New DataSet
    Public sql As String
    Public i As Integer
    Public da As OleDb.OleDbDataAdapter
```

```
End Module
```

Form1

```
Public Class frmForm1
```

```
    Private Sub Label1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label1.Click
```

```
    End Sub
```

```
    Private Sub frmForm1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
```

```
    End Sub
```

```
    Private Sub PictureBox1_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles PictureBox1.Click
```

```
    End Sub
```

```
    Private Sub LinkLabel1_LinkClicked(ByVal sender As System.Object,
ByVal e As System.Windows.Forms.LinkLabelLinkClickedEventArgs) Handles
LinkLabel1.LinkClicked
```

```
        frmForm2.Show()
```

```
        Me.Hide()
```

```
    End Sub
```

```
End Class
```

Form2

```
Public Class frmForm2
```

```
    Private Sub frmForm2_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load  
End Sub
```

```
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
```

```
        sql = "select * from Login"
```

```
        con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=C:\Documents and Settings\Sagar007\My Documents\Visual Studio 2005\Projects\MyThirdProject\MyThirdProject\Bank.mdb"
```

```
        con.Open()
```

```
        da = New OleDb.OleDbDataAdapter(sql, con)
```

```
        da.Fill(ds, "Login")
```

```
        If (txtTextBox1.Text.Trim = "") Then
```

```
            MsgBox("Please Enter User Name")
```

```
            Exit Sub
```

```
        ElseIf (txtTextBox3.Text.Trim = "") Then
```

```
            MsgBox("Please Enter Password")
```

```
            Exit Sub
```

```
        End If
```

```
        For i = 0 To ds.Tables("Login").Rows.Count - 1
```

```
            If (txtTextBox1.Text.Trim =
```

```
ds.Tables("Login").Rows(i).Item(2)) Then
```

```
                If (txtTextBox1.Text.Trim =
```

```
ds.Tables("Login").Rows(i).Item(2)) Then
```

```
                    If (txtTextBox3.Text.Trim =
```

```
ds.Tables("Login").Rows(i).Item(1)) Then
```

```
                        txtTextBox1.Clear()
```

```
                        txtTextBox3.Clear()
```

```
                        MsgBox("Welcome")
```

```
                        frmForm3.Show()
```

```
                        Me.Hide()
```

```
                    Exit For
```

```
                Else
```

```
                    MsgBox("Password Does'nt match")
```

```
                    Exit Sub
```

```
                End If
```

```
            End If
```

```
        Else
```

```
            MsgBox("User Doesn't Exist")
```

```

        txtTextBox1.Clear()
        txtTextBox3.Clear()
        txtTextBox1.Focus()
        Exit For
    End If
Next i

con.Close()
End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
    frmForm1.Show()
    Me.Hide()

End Sub

Private Sub TextBox1_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs)

End Sub

Private Sub txtTextBox1_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txtTextBox1.TextChanged

End Sub

Private Sub btnButton3_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnButton3.Click
    frmForm7.Show()
    Me.Close()

End Sub
End Class

```

Form3

```

Public Class frmForm3

    Private Sub frmForm3_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        lblLabel2.Text = "Welcome" & " " &
ds.Tables("Login").Rows(i).Item(2)
    End Sub

    Private Sub lblLabel2_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles lblLabel2.Click

End Sub

    Private Sub lblLabel1_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles lblLabel1.Click

End Sub

```

```
Private Sub Label1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label1.Click

End Sub

Private Sub Label2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label2.Click

End Sub

Private Sub Label3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label3.Click

End Sub

Private Sub Label4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label4.Click

End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    frmForm4.Show()
    Me.Hide()
    frmForm4.Label2.Text = ds.Tables("Login").Rows(i).Item(2) & " "
& " " & "Your Current Balance is" & " " & ":" &
ds.Tables("Login").Rows(i).Item(3)
    frmForm4.Label3.Text = "Your Account Number is" & " " & ":" &
ds.Tables("Login").Rows(i).Item(0)
End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
    frmForm5.Show()
    Me.Hide()

End Sub

Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
    frmForm6.Show()
    Me.Hide()

End Sub

Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
    frmForm2.Show()
    Me.Hide()

End Sub
End Class
```

Form4

```
Public Class frmForm4

    Private Sub frmForm4_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

        End Sub

    Private Sub Label2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Label2.Click

        End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        frmForm3.Show()
        Me.Hide()
    End Sub
End Class
```

Form5

```
Public Class frmForm5

    Private Sub Label1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Label1.Click

        End Sub

    Private Sub frmForm5_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

        End Sub

    Private Sub Label2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Label2.Click

        End Sub

    Private Sub Label3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Label3.Click

        End Sub

    Private Sub TextBox1_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles TextBox1.TextChanged

        End Sub

    Private Sub TextBox2_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles TextBox2.TextChanged

        End Sub
End Class
```

```

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Dim Acc_Bal As String
    Dim New_Bal As String
    Dim str As String
    Dim cmd As New OleDb.OleDbCommand
    'Dim cb As New OleDb.OleDbCommandBuilder(da)
    con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=C:\Documents and Settings\Sagar007\My Documents\Visual Studio
2005\Projects\MyThirdProject\MyThirdProject\Bank.mdb"
    con.Open()
    Acc_Bal = ds.Tables("Login").Rows(i).Item(3)
    TextBox2.Text = Acc_Bal + Double.Parse(TextBox1.Text)
    New_Bal = TextBox2.Text

    ds.Tables("Login").Rows(i).Item("Account_Bal") = New_Bal
    str = "update Login set Account_Bal='" & New_Bal & "' where
Account_No=" & i & " "
    cmd.Connection = con
    cmd.CommandText = str

    cmd.ExecuteNonQuery()
    con.Close()
    'da.Update(ds, "Login")
End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
    frmForm3.Show()
    Me.Hide()
End Sub
End Class

```

Form6

```

Public Class frmForm6

    Private Sub frmForm6_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load

        End Sub

        Private Sub Label2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label2.Click

            End Sub

        Private Sub Label3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label3.Click

            End Sub

        Private Sub TextBox1_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles TextBox1.TextChanged

            End Sub

```

```

Private Sub TextBox2_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles TextBox2.TextChanged

End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
Dim Cur_Bal As String
Dim WithDraw_Bal As String
Dim New_Bal As String
Dim str As String
Dim cmd As New OleDb.OleDbCommand
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=C:\Documents and Settings\Sagar007\My Documents\Visual Studio
2005\Projects\MyThirdProject\MyThirdProject\Bank.mdb"
con.Open()
Cur_Bal = ds.Tables("Login").Rows(i).Item(3)
WithDraw_Bal = TextBox1.Text
If (Cur_Bal > WithDraw_Bal) Then
New_Bal = Cur_Bal - WithDraw_Bal
TextBox2.Text = New_Bal
'Dim cb As New OleDb.OleDbCommandBuilder(da)
ds.Tables("Login").Rows(i).Item(3) = New_Bal
str = "update Login set Account_Bal= '" & New_Bal & "'
where Account_No=" & i & " "
cmd.Connection = con
cmd.CommandText = str

cmd.ExecuteNonQuery()

'da.Update(ds, "Login")
Else
MsgBox("Withdraw Amount is Greater Than Current Balance")
End If
con.Close()
End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
frmForm3.Show()
Me.Hide()

End Sub
End Class

```

Form7

```
Imports System.Data.DataRow
Imports System.Data.DataRowBuilder
```

```
Public Class frmForm7
```

```
    Private Sub TextBox3_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles TextBox3.TextChanged
```

```
    End Sub
```

```
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
```

```
        If (TextBox1.Text = "") Then
            MsgBox("Enter User Name")
            Exit Sub
```

```
        ElseIf (TextBox2.Text = "") Then
            MsgBox("Enter Password")
            Exit Sub
```

```
        ElseIf (TextBox3.Text = "") Then
            MsgBox("Enter Confirm Password")
            Exit Sub
```

```
        End If
```

```
        sql = "select * from Login"
```

```
        con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=C:\Documents and Settings\Sagar007\My Documents\Visual Studio
2005\Projects\MyThirdProject\MyThirdProject\Bank.mdb"
```

```
        con.Open()
```

```
        da = New OleDb.OleDbDataAdapter(sql, con)
```

```
        da.Fill(ds, "Login")
```

```
        For i = 0 To ds.Tables("Login").Rows.Count - 1
```

```
            If (TextBox1.Text.Trim =
ds.Tables("Login").Rows(i).Item(2)) Then
                MsgBox("User Exist Enter Some Other Name")
```

```
                TextBox1.Clear()
```

```
                TextBox2.Clear()
```

```
                TextBox3.Clear()
```

```
                TextBox4.Clear()
```

```
                TextBox1.Focus()
```

```
                con.Close()
```

```
            Exit Sub
```

```
        End If
```

```
    Next i
```

```
    If (TextBox2.Text = TextBox3.Text) Then
```

```
        Dim str As String
```

```
        Dim cmd As New OleDb.OleDbCommand
```

```
        str = "select count(User_Name) from Login"
```

```
        cmd.CommandText = str
```

```
        cmd.Connection = con
```

```
        Dim c As Integer
```

```

Dim dr As Data.IDataReader = cmd.ExecuteReader()
While dr.Read
    c = Val(dr.Item(0).ToString)

End While
dr.Close()
If c = 0 Then
    c = 1
Else
    c = c + 1
End If
str = "insert into Login values(" & c & " ,'" &
TextBox2.Text & "','" & TextBox1.Text & "','" &
Double.Parse(TextBox4.Text) & ")"
cmd.Connection = con
cmd.CommandText = str

cmd.ExecuteNonQuery()

MsgBox("New Record added to the Database")
TextBox1.Clear()
TextBox2.Clear()
TextBox3.Clear()
TextBox4.Clear()
con.Close()
Else
MsgBox("Password Doesn't Match")
TextBox2.Clear()
TextBox3.Clear()
TextBox2.Focus()
End If

End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
    frmForm2.Show()
    Me.Hide()

End Sub

Private Sub TextBox4_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles TextBox4.TextChanged

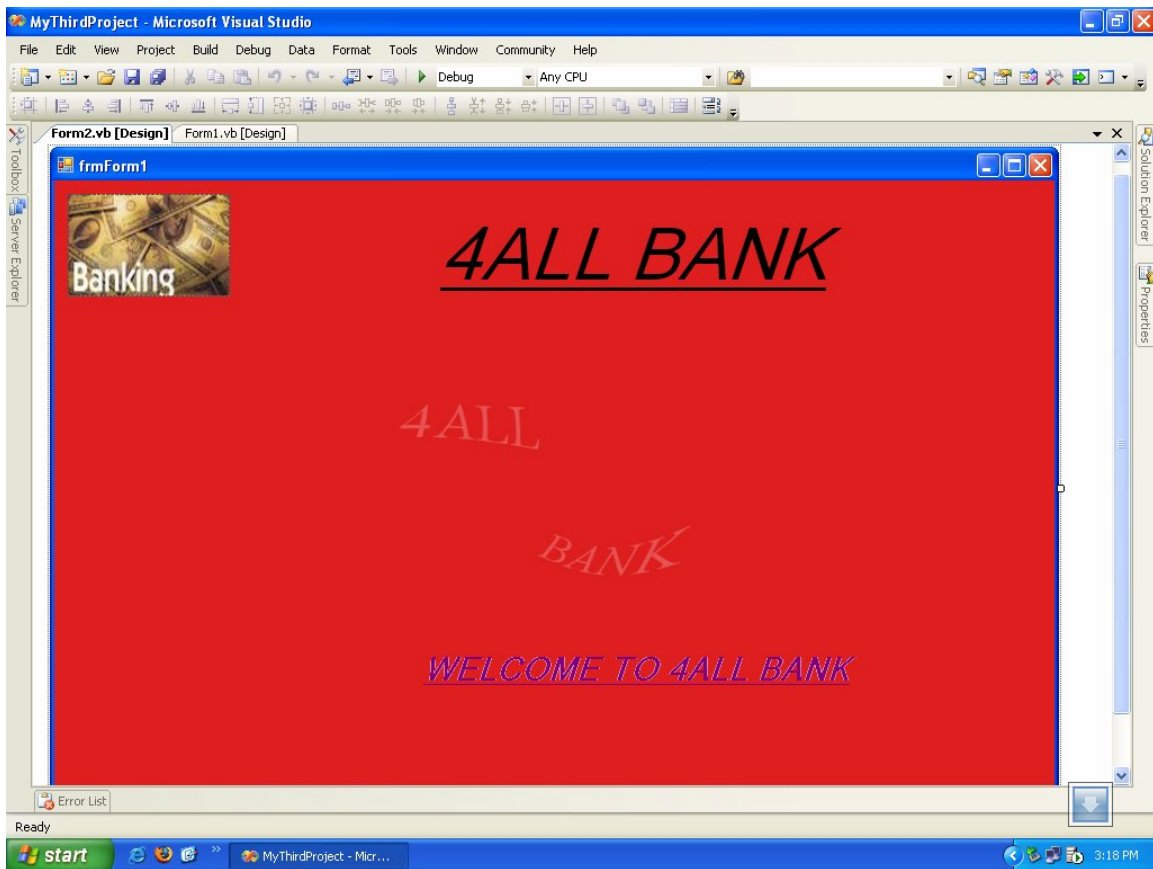
End Sub

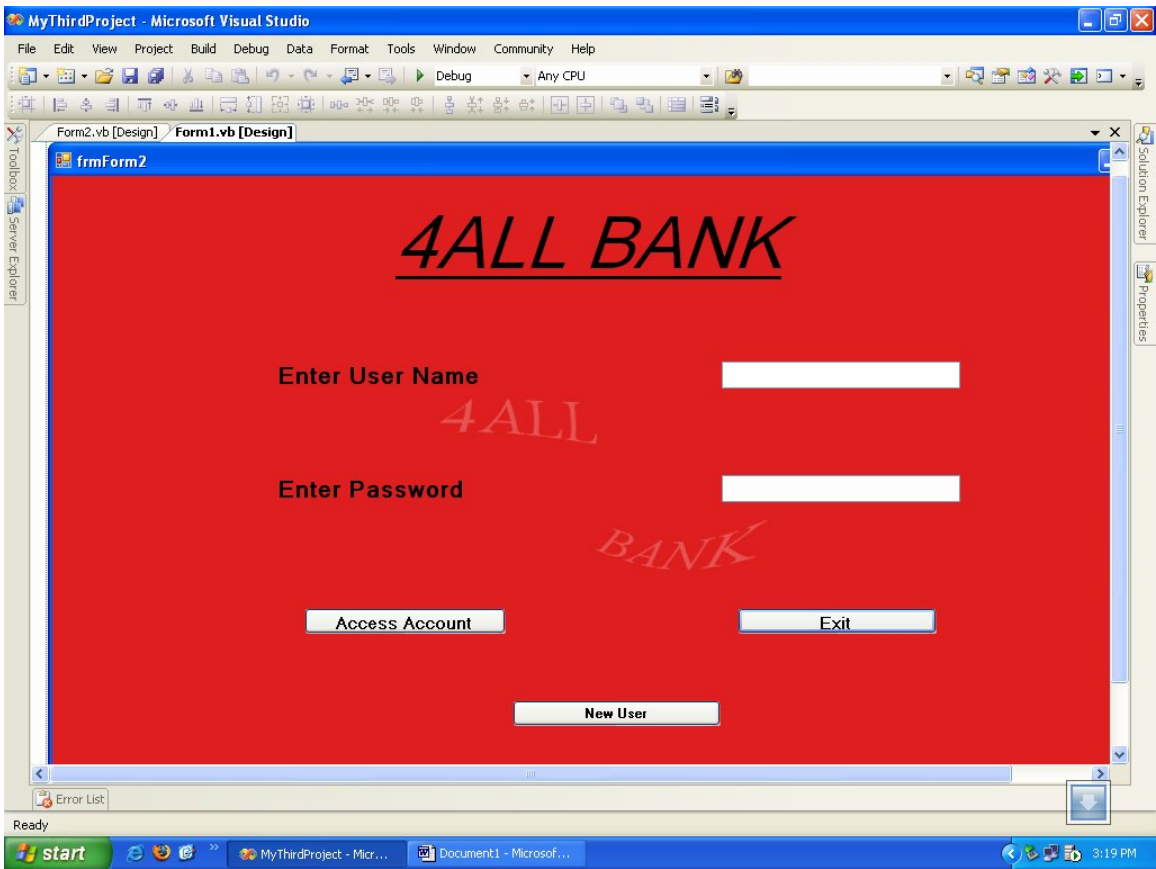
Private Sub frmForm7_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load

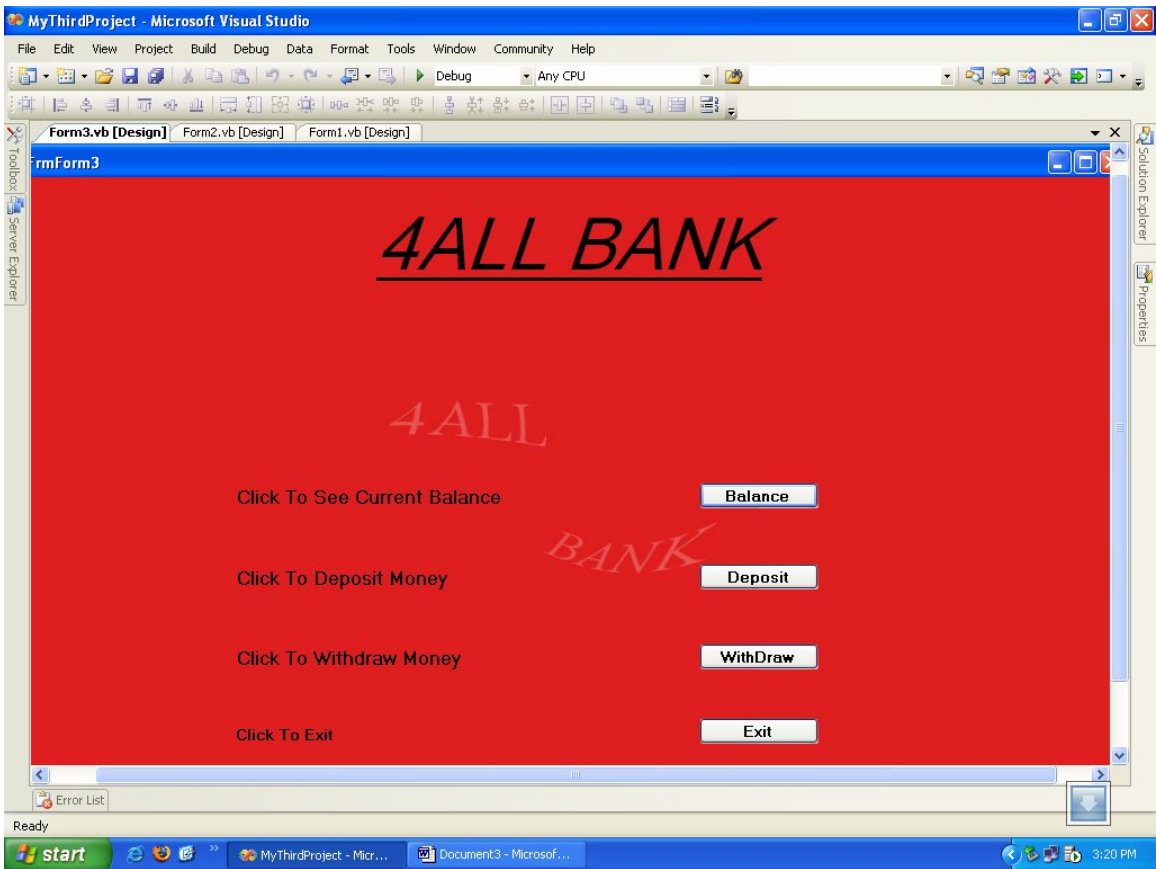
End Sub
End Class

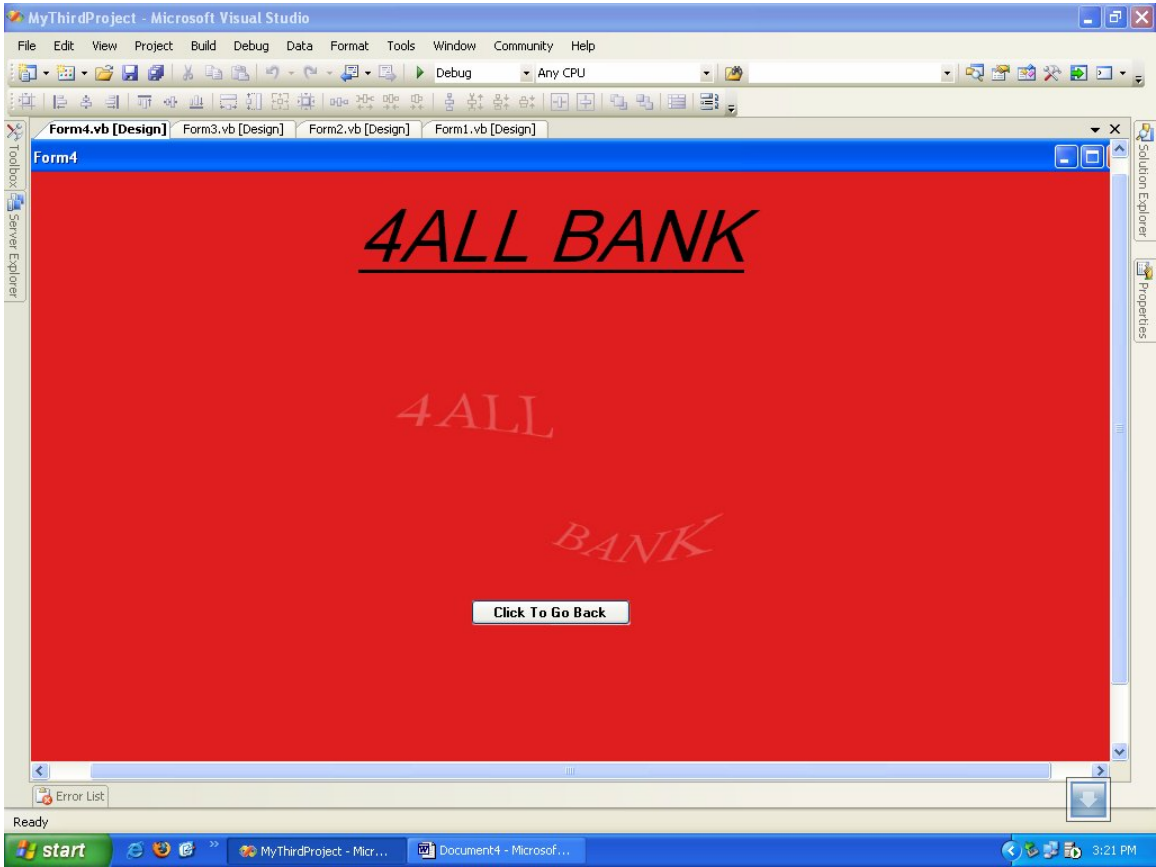
```

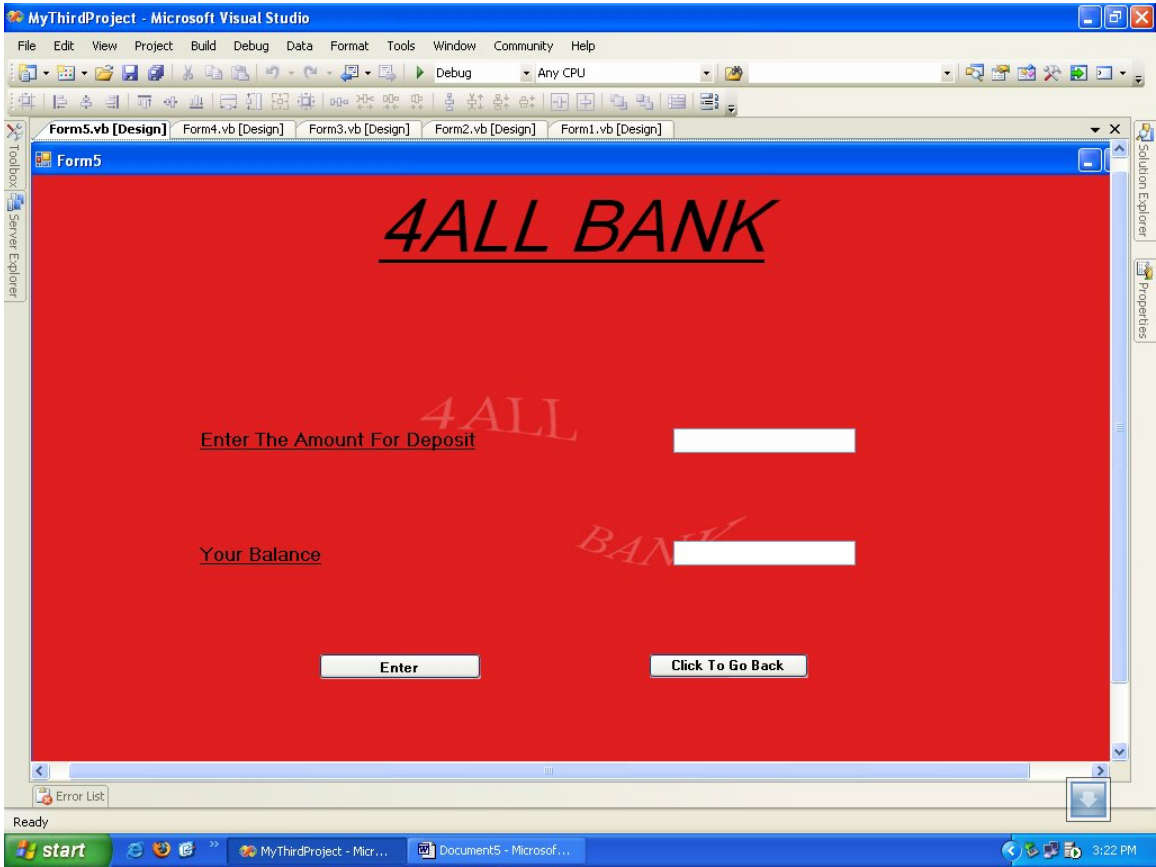
FORM LAYOUT

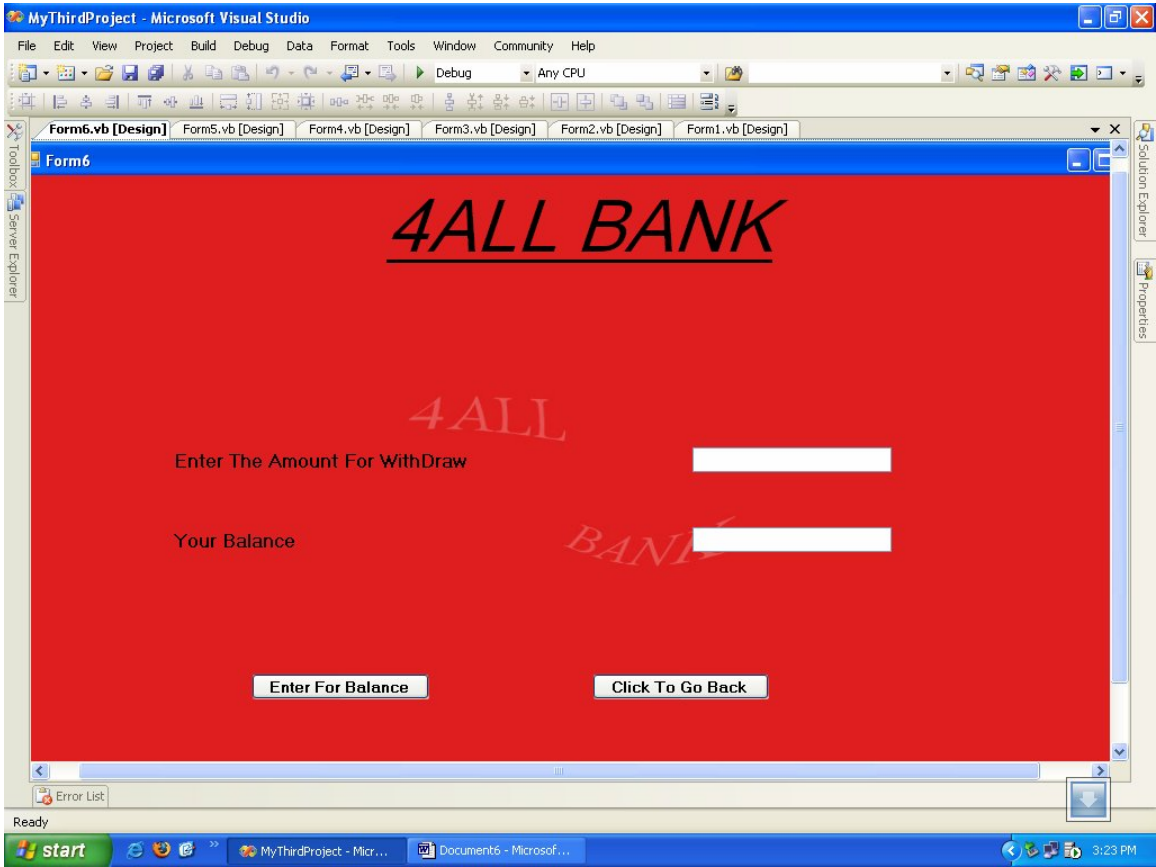


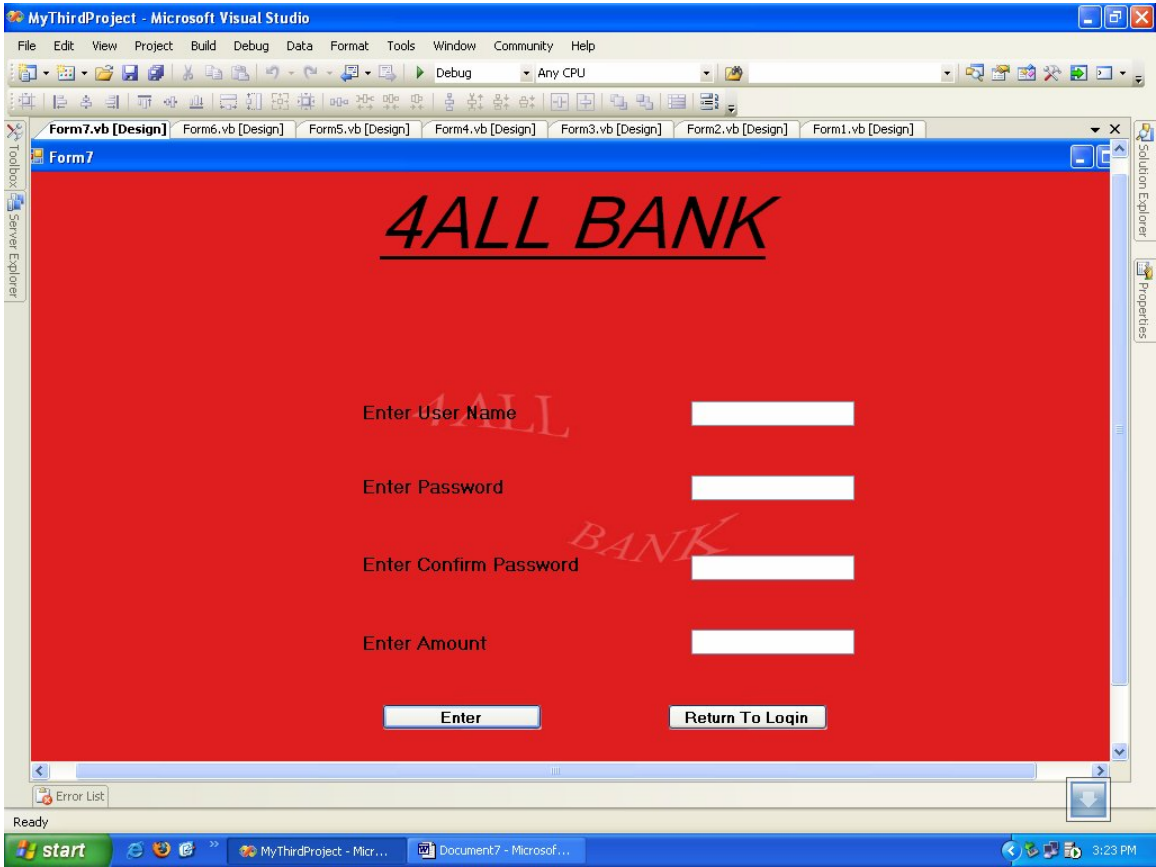












CONCLUSION

This project developed, incorporated all the activities involved in the browsing centre. It provides all necessary information to the management as well as the customer with the use of this system; the user can simply sit in front of the system and monitor all the activities without any physical movement of the file. Management can service the customers request best in time.

The system provides quickly and valuable information. These modules have been integrated for effective use of the management for future forecasting and for the current need.

SCOPE FOR FURTHER DEVELOPMENT:-

The system can be designed for further enhancement .This could also be developed according to the growing needs of the customer.